

RECOLAB: LABORATORIO REMOTO DE CONTROL UTILIZANDO MATLAB Y SIMULINK

Luis M. Jiménez, Rafael Puerto, Óscar Reinoso
César Fernández, Ramón Neco

*Dpto. Ingeniería de Sistemas Industriales, Universidad
Miguel Hernández, Elche (Alicante), 03202 Spain*

Resumen: En este artículo se presenta una arquitectura general aplicable a la ejecución remota en tiempo real vía internet de controladores sobre procesos físicos. Esta aplicación ha sido desarrollada utilizando la plataforma Matlab/Simulink (Mathworks, 2004). La motivación de este trabajo está basada en la limitada disponibilidad de sistemas físicos reales en los laboratorios para realizar prácticas en cursos de ingeniería de control. De este modo, se permite el uso compartido de varios procesos físicos, disponibles en el laboratorio, de forma remota. Asimismo se presentan algunos ejemplos que muestran la validez y aplicabilidad de la arquitectura presentada.

Copyright ©2005 CEA-IFAC

Palabras Clave: Sistemas de Control en Tiempo Real, Control Remoto Vía Internet, Educación en Control.

1. INTRODUCCIÓN

El trabajo presentado en este artículo, pretende proporcionar una arquitectura general para la ejecución remota en tiempo real de esquemas de control sobre sistemas físicos. La motivación de este trabajo ha sido, por un lado, la carencia de suficientes maquetas o sistemas reales en las prácticas de teoría de control; y por otro, la limitada disponibilidad temporal de laboratorios donde, los estudiantes que cursan materias de automática y control de procesos, puedan desarrollar los conceptos vistos en teoría dentro de un marco práctico adecuado. Por tanto, la presente aplicación se centra en el control en tiempo real de sistemas físicos de forma remota a través de Internet, utilizando para ello Matlab como plataforma de desarrollo. Esta aplicación permite al estudiante tanto simular como ejecutar en tiempo real, un esquema de control sobre un determinado

sistema físico. De esta forma, el estudiante puede desarrollar estas actividades sin estar presente en el laboratorio donde esté ubicado el sistema real. La aplicación devuelve al usuario toda la información asociada a la simulación o ejecución solicitada, junto a los gráficos, datos y variables más relevantes.

Durante los últimos años el uso de las capacidades que proporciona Internet ha posibilitado la realización de diversas herramientas que pueden mejorar significativamente la calidad de los procesos de enseñanza y aprendizaje. En enseñanzas técnicas además de los contenidos teóricos es fundamental la realización de prácticas con equipos reales de forma que el alumno asimile de forma adecuada los conceptos teóricos. Por este motivo desde principios de los 90 educadores en ingeniería han demostrado la capacidad de realizar experimentos en laboratorios docentes mediante el uso de Internet y tecnologías web. Así, la educación a

distancia mediante el uso de laboratorios remotos a través de Internet es un tema de total actualidad como consecuencia del enorme potencial que supone para mejorar la calidad del proceso enseñanza-aprendizaje (Ausserhofer, 1999).

Un laboratorio a distancia permite a los usuarios realizar experimentos desde una posición remota. Los laboratorios a distancia pueden dividirse en dos clases: laboratorios virtuales y laboratorios remotos (Casini *et al.*, 2003). En los primeros, los estudiantes pueden ejecutar simulaciones remotas junto con posibles animaciones. En los segundos, los estudiantes hacen uso remoto de los equipos reales disponibles en los laboratorios. Un estudio mucho más detallado entre laboratorios remotos y virtuales se encuentra en (Exel *et al.*, 2000). Sin duda la realización de prácticas con sistemas físicos reales permite al estudiante adquirir y afianzar los conocimientos de una forma más eficiente que al realizar únicamente experimentos simulados. Por este motivo algunos autores proponen el uso, en la medida de lo posible, de laboratorios remotos para afianzar los contenidos teóricos desarrollados, especialmente en el caso de estudiantes de ingeniería (Exel *et al.*, 2000). A través de estos laboratorios remotos, los usuarios pueden realizar y planificar experimentos, analizar los datos experimentales alcanzados como si estuvieran físicamente presentes en el laboratorio en cualquier momento del día.

Son numerosas las disciplinas en las que se han propuesto la utilización de laboratorios virtuales y remotos para la realización de prácticas (Molly, 2000). Así, en (Molly and Robby, 2000) se muestran algunas líneas básicas sobre las características y capacidades que debe tener un sistema de control de procesos basado en Internet. En (Guimaraes *et al.*, 2003) un laboratorio virtual accesible a través de internet permite a un usuario remoto manipular un robot móvil en un modo de interacción adecuado a su nivel de experiencia. También en (Candelas *et al.*, 2004) se presentan los resultados de aceptación en el uso de un laboratorio virtual que permite a los estudiantes realizar prácticas de robótica de forma remota. Asimismo se proponen otros laboratorios a distancia para manipular robots (Hirukawa and Hara, 2000). En (Sebastian *et al.*, 2003), se presenta un entorno para la realización de prácticas de visión por computador a través de Internet que incluye la manipulación de diversos dispositivos, pero sin el uso de control en tiempo real. Para ello, los autores utilizan herramientas software de distinta naturaleza y proveedores que incrementan el tiempo de desarrollo y el coste de mantenimiento.

En particular, en la enseñanza de la disciplina de control las experiencias prácticas realizadas con sistemas físicos son significativamente impor-

tantes en relación con la enseñanza de sistemas de control (Kleir *et al.*, 1996). En este sentido, recientemente se han propuesto diferentes laboratorios de sistemas de control para la realización de prácticas mediante acceso remoto (Pastor *et al.*, 2001), (Puerto *et al.*, 2001). El desarrollo de laboratorios de control puede resultar costoso tanto en tiempo como en dinero para la mayor parte de las instituciones. Una de las motivaciones principales en el desarrollo de laboratorios de control a distancia es, por un lado la ausencia de modelos a escala sobre sistemas físicos que tengan las mismas propiedades que éstos, y por otro la escasa disponibilidad de tiempo en los laboratorios donde la presencia de estos equipos es posible. Muchos de los resultados presentados hasta el momento combinan diferentes desarrollos que posibilitan la realización práctica de experimentos de sistemas de control a través de Internet (Puerto *et al.*, 2004) (Schmid, 2001), (Overstreet and Tzes, 1999).

Una de las características básicas del sistema propuesto en este artículo radica en el uso de unas herramientas de diseño conocidas por el estudiante para la realización de las prácticas en la disciplina de ingeniería de sistemas como son Matlab y Simulink. Estas herramientas permiten a los estudiantes y a los educadores concentrarse en los aspectos de diseño e implementación de sistemas de control, en lugar de la costosa programación a bajo nivel (Dixon *et al.*, 2001) (Dixon *et al.*, 2002).

Schmid (Schmid, 2001) presenta un laboratorio virtual que utiliza Matlab/Simulink para simulaciones. En este caso la animación interactiva de experimentos de control se realiza mediante modelos VRLM de las maquetas de laboratorio. Bonivento (Bonivento *et al.*, 2002) propone un laboratorio remoto de control, basado en Matlab/Simulink pero con un software específico desarrollado en Visual C++ para interactuar con los modelos físicos. En (Díez *et al.*, 2001) se presenta una herramienta basada en Matlab Web Server para la ejecución de aplicaciones Matlab utilizando modelos virtuales. Overstreet (Overstreet and Tzes, 1999) presenta una plataforma en la cual el usuario puede ejecutar un controlador sobre un experimento que reside no solo en el servidor, si no también en el cliente, compilándolo y ejecutándolo en la máquina del usuario. Finalmente Casini (Casini *et al.*, 2003), propone el uso de un laboratorio remoto de control utilizando Matlab y Real-Time Workshop (RTW), que permite la transformación de un modelo Simulink a código C. Sin embargo, en esta plataforma, es preciso código específico para enlazar el código fuente generado por RTW, haciendo posible establecer una conexión TCP.

Por estos motivos, se ha elegido la plataforma Matlab / Simulink (con algunos toolboxes adi-

cionales) como herramienta de desarrollo de aplicación propuesta por varias razones: en primer lugar, Matlab, Simulink y los toolboxes necesarios, constituyen una plataforma fiable y ampliamente utilizada, que además goza de un adecuado soporte técnico así como una extensa aplicación a la enseñanza en los cursos de control. En segundo lugar, el tiempo necesario para el desarrollo de un prototipo de laboratorio es bastante inferior al necesitado por otras herramientas y/o plataformas. Por otra parte, proporciona herramientas para la ejecución remota de programas, así como el control en tiempo real de un sistema físico, a través de un sistema de adquisición de datos. Por último, y no menos importante, es la gran cantidad de investigadores que utilizan esta plataforma como herramienta de desarrollo para simulación y para el control de sistemas reales.

2. DESCRIPCIÓN GENERAL DEL SISTEMA

El esquema general de la arquitectura de la aplicación desarrollada, denominada RECOLAB (REmote Control LABoratory), se muestra en la Figura 1. En este diagrama, los elementos hardware y software están divididos en dos bloques principales: La zona local donde el usuario trabaja, y la zona remota donde se sitúan el sistema físico y todos los elementos de control (tarjeta de adquisición de datos, sensores, actuadores, etc.). De forma detallada se describen a continuación los elementos de ambas zonas:

- Zona local:
 - Computador con conexión a Internet y un cliente HTTP 4.0. La aplicación RECOLAB está optimizada para Internet Explorer 6 y Netscape 7 con una resolución mínima de 800x600 puntos.
- Zona remota:
 - Conexión a Internet de alta velocidad.
 - Computador Servidor: el actual sistema consiste en un PC Pentium III (800 MHz) con 256 MB de RAM, que proporciona la suficiente potencia para ejecutar el servidor web y las aplicaciones de tiempo real (proporcionadas por la plataforma Matlab).
 - Sistema de adquisición de datos: Tarjeta de adquisición NI 6024E con entradas y salidas analógicas y digitales.
 - Sistema físico a controlar: Motor CC modelo Feedback 33-002. Esta maqueta de laboratorio está integrada por un motor de corriente continua, amplificador de potencia, tacómetro, encoder absoluto, potenciómetro y freno magnético.
 - Sistema de captación de imágenes y servidor de vídeo: formado por una cámara

Sony EVI-D31, y un servidor de vídeo Axis 2400. Este servidor de vídeo, basado en Linux empotrado, es capaz de realizar la captura de hasta cuatro cámaras en color, el control de movimiento de las mismas, y servir tanto imágenes estáticas como vídeo continuo.

- Sistema Operativo que permita establecer directivas de seguridad. Actualmente Windows 2000 Professional.
- Servidor HTTP Apache v.1.3.31 con los módulos PHP 4.3.4 y SSL *Secure Socket Layer*. Este servidor permite la comunicación con los ordenadores cliente utilizando el protocolo HTTP.
- MATLAB R12 con SIMULINK V. 4.1: ejecuta la aplicación que hace posible el control en tiempo real del sistema y la generación de los ficheros de resultados.
- Matlab Web Server V. 1.2.1: este toolbox permite utilizar las capacidades gráficas y matemáticas de Matlab a través de una página web.
- Real – Time Windows Target Toolbox V. 2.1: Esta toolbox permite ejecutar esquemas Simulink en tiempo real. Para ello, proporciona los bloques necesarios para la comunicación con la tarjeta de adquisición de datos.
- Control System Toolbox 5.1: herramienta básica de análisis y diseño de sistemas de control.

2.1 Funcionamiento Interno de RECOLAB

La aplicación RECOLAB tiene dos aspectos claramente diferenciados:

1. Aplicación Web: incluye la comunicación cliente – servidor utilizando los protocolos HTTP/HTML, el control de acceso de usuarios y la aplicación CGI (Common Gateway Interface) principal. CGI es un interfaz estándar que permite la comunicación con aplicaciones externas al servidor Web. La aplicación CGI de RECOLAB se encarga del acceso a los recursos y la comunicación entre el servidor Apache y las aplicaciones Matlab y Matlab Web Server.
2. Aplicación de control del sistema físico: realiza la ejecución en tiempo real de un esquema de control sobre un sistema físico determinado.

Se han utilizado diferentes lenguajes de programación y herramientas de desarrollo para cada una de las dos partes comentadas. Las tareas no críticas, interfaz de usuario, seguridad de acceso y compartición de recursos, han sido codificadas en PHP(v.4.3.2) ejecutándose sobre un servidor

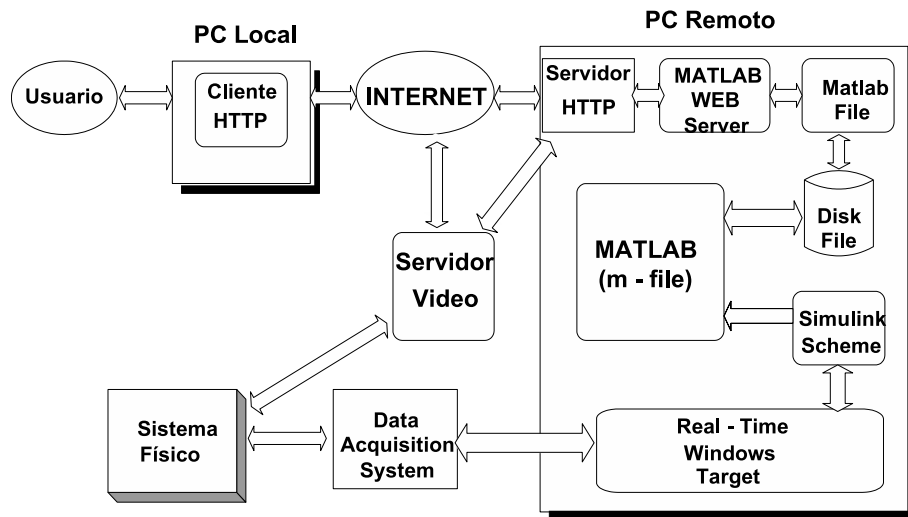


Figura 1. Arquitectura General de RECOLAB.

HTTP Apache (Apache, 2004). PHP (PHP, 2004) es un lenguaje interpretado muy popular que ha sido escogido debido a su carácter de lenguaje abierto y ampliamente soportado por la mayoría de servidores Web y sistemas operativos en múltiples plataformas. Asimismo, incluye una extensa librería que soporta la mayoría de protocolos de red y el acceso a bases de datos. La aplicación PHP se ejecuta en un servidor web, por tanto, presenta un entorno controlado para el programador y puede comunicarse con cualquier otro proceso que esté ejecutándose en el servidor (la aplicación Matlab en nuestro caso).

Por supuesto, PHP es un lenguaje interpretado no indicado para el desarrollo de aplicaciones en tiempo real. El núcleo de tiempo real de RECOLAB (el control realimentado del sistema físico) ha sido desarrollado en Simulink con Real - Time Windows Target (Mathworks, 2004).

2.1.1. Interfaz de Usuario de RECOLAB. El interfaz de usuario de RECOLAB, está basado en el lenguaje estándar HTML 4.0. El código HTML es generado dinámicamente mediante PHP. El diseño de la página HTML está separado de los contenidos para permitir una configuración flexible de la aplicación RECOLAB. Todo el formato de datos está basado en hojas de estilo CSS (*Cascade Style Sheets*) y funciones predefinidas PHP para la creación de los formularios. El contenido de las páginas se almacena en ficheros de texto haciendo fácil y flexible el añadir nuevos sistemas físicos y sistemas de control sin ninguna modificación en el código fuente de la aplicación.

El interfaz de usuario mostrado en la figura 2 está basado en formularios HTML generados dinámicamente en PHP a partir de los ficheros de configuración. La actualización interactiva de estos formularios se realiza a través de llamadas

CGI al servidor y no a través de código en el navegador del cliente. De esta forma, la seguridad en el acceso al sistema se puede manejar de forma más eficiente.

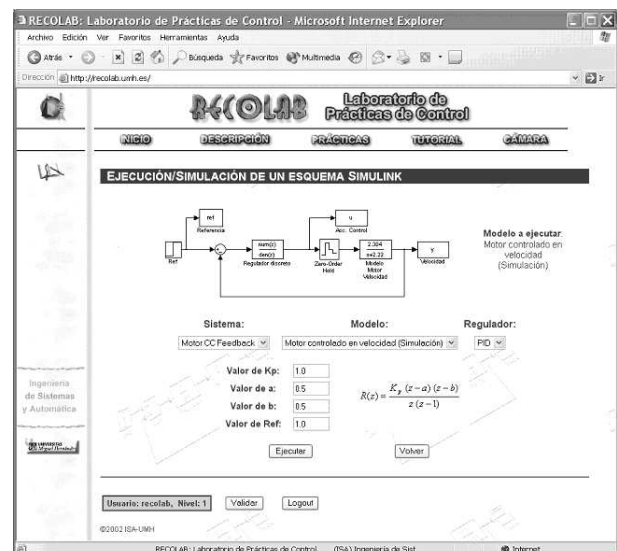


Figura 2. Control de Acceso en RECOLAB.

Los diferentes esquemas Simulink de control disponibles pueden ejecutarse tanto en simulación, como sobre el sistema físico real mediante una tarjeta de adquisición de datos. El usuario puede elegir diferentes estrategias de control y tipos de regulador, seleccionando los parámetros de acuerdo a la experimento a realizar.

Cuando termina la ejecución solicitada, el usuario puede descargar el conjunto completo de variables medidas del proceso. La página de resultados muestra de forma gráfica un extracto de esta información (la salida del sistema) (figura 3).

2.1.2. Sistema de Control de Acceso de RECOLAB. La mayor parte del portal web RECO-

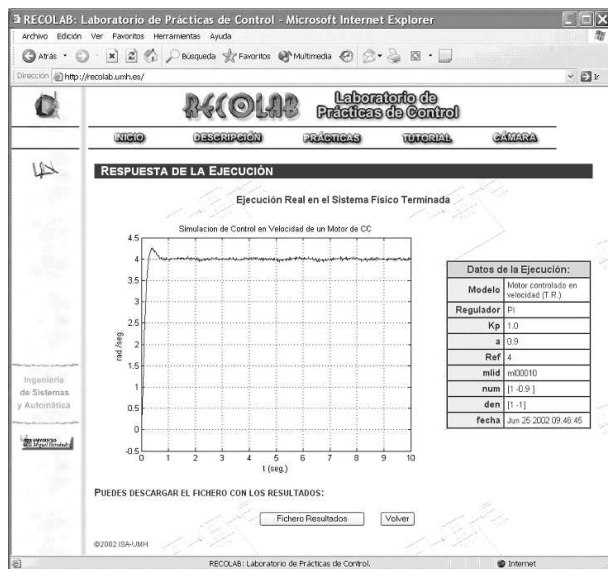


Figura 3. Página de Resultados

LAB puede ser accedido desde Internet sin ninguna restricción, en la medida que se trata de una herramienta educativa abierta. De esta forma puede ser utilizada por cualquier estudiante interesado en el control de procesos. Así, se posibilita un acceso sin ningún tipo de restricción para realizar las simulaciones mediante esquemas Simulink de diferentes sistemas físicos y modelos (control en posición o en velocidad). Mediante estas simulaciones cualquier usuario puede seleccionar el tipo de regulador más adecuado así como los parámetros del mismo y realizar la simulación del esquema en bucle cerrado. Tras la simulación, el sistema devuelve al usuario la respuesta del proceso ante la señal de entrada seleccionada y con el regulador empleado. Existe además la posibilidad de que el usuario almacene estos datos en un fichero con formato Matlab (*.mat). Así, el usuario tras enviar la petición al servidor remoto y recibir los resultados, puede realizar un análisis más exhaustivo en su computador local.

Por otra parte, existen opciones dentro del sistema que sólo puede ser accedido por usuarios validados. Estas zonas guardan relación con la utilización del sistema físico. El punto más crítico en RECOLAB es la ejecución en tiempo real de esquemas Simulink sobre sistemas físicos. Para tener un uso más eficiente de los recursos del laboratorio, este proceso precisa de una validación por parte del usuario. Así, tanto para tener acceso a la ejecución de un esquema de control en tiempo real sobre el sistema físico seleccionado como para acceder al servidor de vídeo es preciso una validación del usuario ya que estos elementos tienen acceso restringido.

El sistema de validación utilizado por RECOLAB no está basado en un sistema estándar proporcionado por el servidor web, ya que éstos están

orientados al control de recursos web y no al control de recursos de la aplicación. En su lugar, el sistema de validación utilizado está codificado en PHP permitiendo su integración con la aplicación principal. Las opciones mostradas por la aplicación web dependen de la validación realizada por el usuario. De esta forma, no es preciso mantener diferentes páginas para usuarios validados o no. Si un usuario solicita una opción que precisa validación (por ejemplo una ejecución en tiempo real), el código PHP solicita dicha validación.

El proceso de validación se basa en el nombre de un usuario y un password. El control de acceso a los recursos, en cambio, se basa en el nombre del usuario y un nivel de prioridad. La mayoría de los recursos en RECOLAB utilizan un control de accesos mediante niveles. El sistema utiliza actualmente cuatro niveles: *administración*, *profesor*, *usuario* e *invitado*. Este último nivel permite la ejecución de simulaciones sobre RECOLAB. Los estudiantes registrados disponen de una cuenta de nivel *usuario* que permite utilizar todas las funciones de ejecución en tiempo real y visualización de la imagen del sistema físico. Los niveles de *administración* y *profesor* permiten modificar parámetros adicionales del sistema.

2.1.3. La Aplicación CGI. Esta aplicación enlaza los datos introducidos por el usuario y el programa de control de Matlab. Sincroniza los diferentes módulos y devuelve los resultados al usuario. Para realizar esta tarea, utiliza una aplicación intermedia: Matlab Web Server Toolbox.

Este toolbox permite, de una forma sencilla, adaptar los datos proporcionados por el usuario a Matlab. Asimismo, puede ejecutar funciones Matlab y generar páginas web gráficas con los resultados. Desafortunadamente, dicho toolbox está orientado a la implementación de laboratorios virtuales, pero no a laboratorios físicos remotos. No dispone de un mecanismo para comunicarse con Real - Time Windows Target para poder ejecutar aplicaciones de control en tiempo real. Por esta razón, Matlab Web Server se utiliza únicamente para la tarea de adaptar los datos de la interfaz de usuario a Matlab. El resto de las tareas de gestión no críticas han sido desarrolladas en PHP, utilizando ficheros como mecanismo básico para comunicar los datos precisos para ejecutar el control en tiempo real (parámetros del controlador, etc.) y recuperar los resultados obtenidos una vez que el experimento ha sido realizado. En la próxima sección se comenta como se realiza esta comunicación entre los diferentes módulos.

2.1.4. Aplicación de Control en Tiempo Real. Esta aplicación ejecuta el control realimentado en tiempo real utilizando el esquema de control

especificado por el usuario sobre el sistema físico. La aplicación ha sido desarrollada utilizando las siguientes herramientas: Matlab, Simulink y Real-Time Windows Target (Mathworks, 2004). Esta última herramienta permite generar el código de tiempo real para ejecutar esquemas Simulink utilizando un sistema de adquisición de datos (y por tanto, sobre un sistema físico conectado a él). El hecho de ejecutar un esquema Simulink directamente sobre el sistema físico, proporciona la ventaja añadida de fácil implementación y posterior modificación del esquema de control propuesto.

2.1.5. Comunicación entre las Tareas de Tiempo Real e Interfaz de Usuario. Como se ha descrito previamente, la aplicación está formada por dos subaplicaciones que ejecutan las tareas de tiempo real y de interfaz de usuario. Es preciso por tanto, un método eficiente de comunicación y sincronización de ambos procesos.

Para resolver este problema, la aplicación codificada en PHP lanza la tarea de control en tiempo real implementada en Matlab con los parámetros especificados por el usuario y sincroniza su ejecución mediante un mecanismo de semáforos basado en ficheros. El procedimiento completo implementado para ejecutar el esquema Simulink, se detalla a continuación:

1. El usuario (cliente) se conecta al servidor RECOLAB y elige, a través del interfaz de usuario, la ejecución o simulación, de un esquema Simulink determinado. Por ejemplo, el usuario puede seleccionar el tipo y los parámetros del controlador, cambiar los valores de los coeficientes de los polinomios del numerador y denominador de su función de transferencia o la matriz de ganancia de realimentación de estado.
2. La aplicación CGI asociada a la página de la interfaz de usuario, ejecuta la llamada a Matlab Web Server.
3. Matlab Web Server almacena los datos necesarios en un fichero Matlab (.mat) y devuelve el control a la aplicación PHP. Un mecanismo de sincronización basado en ficheros mantiene la coherencia de dichos ficheros de datos permitiendo varias conexiones concurrentes.
4. La aplicación CGI toma el control de la ejecución leyendo los datos asociados al experimento y lanza la aplicación Matlab, encargándose asimismo de la compartición del sistema físico entre varias conexiones concurrentes. Adicionalmente supervisa la ejecución y detecta cualquier problema que pueda suceder. Tras chequear los privilegios de acceso del usuario, lee los datos de configuración del experimento y espera al semáforo de sincronización entre conexiones (espera a

que ninguna otra sesión esté utilizando los recursos del laboratorio). Una vez abierto el semáforo, lanza una sesión Matlab en background. Esta última acción provoca la ejecución de un fichero Matlab específico. Este programa lee el fichero de configuración (escrito por Matlab Web Server), carga el fichero Simulink adecuado, adaptando sus parámetros (seleccionados por el usuario), y lanza la simulación o la ejecución en tiempo real.

5. Una vez terminada la simulación o la ejecución en tiempo real, todos los resultados son almacenados (incluyendo figuras con respuesta temporal de las señales pertinentes) y la aplicación PHP es reactivada por medio del mecanismo de semáforos. Una vez reactivada, esta aplicación lee los datos de los resultados de la ejecución del experimento y genera la página HTML con los gráficos obtenidos. En esta misma página, el usuario puede descargar el fichero de datos generado durante el experimento (fichero .mat) para un análisis posterior en un computador local.

Este procedimiento se muestra esquemáticamente en la figura 4.

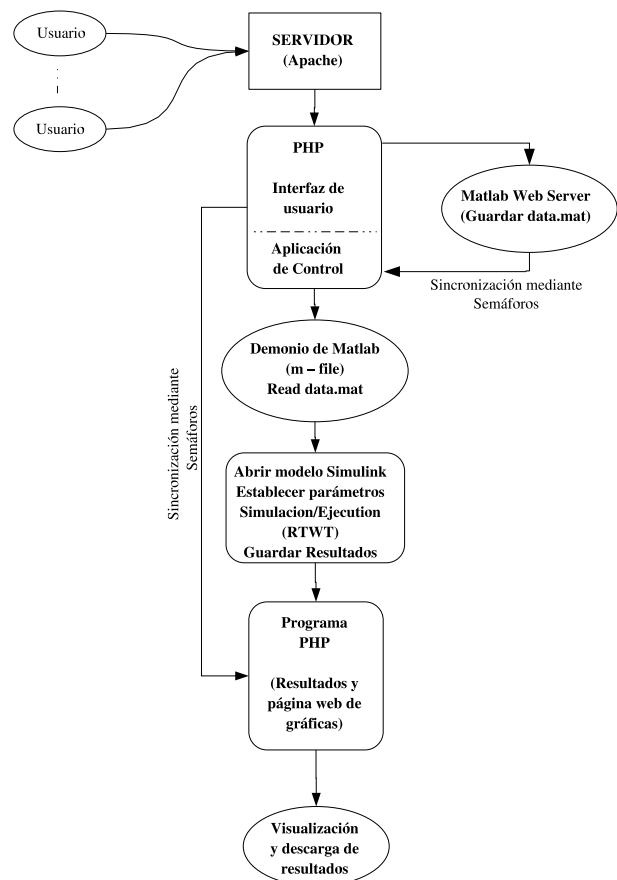


Figura 4. Funcionamiento de RECOLAB.

Cabe resaltar que sólo puede realizarse al mismo tiempo una ejecución, en tanto que se utilice el sistema físico y los recursos de tiempo real. Este es

el motivo por el que se ha implementado el mecanismo de control mediante semáforos. Cuando se acepta más de una petición por parte del servidor web Apache, éstas se encolan hasta que la ejecución previa termina o expira un temporizador. Obviamente, sólo un usuario puede ejecutar un experimento sobre el sistema físico en tiempo real a la vez. Cuando sólo se solicita una simulación (el sistema permite también realizar simulaciones de esquemas Simulink, de hecho se recomienda antes de solicitar la ejecución en tiempo real), ésta puede ser realizada concurrentemente con otras simulaciones.

3. EJEMPLO

En esta sección se presenta un ejemplo de utilización del sistema RECOLAB (RECOLAB, 2004). La tarea a realizar consiste en ejecutar, en tiempo real y a través de Internet, el control (en este caso mediante un regulador PID) de un servomotor de corriente continua. Se ha elegido este sistema físico por sus características: fácil identificación, comportamiento lineal, etc. Sin embargo, la arquitectura del sistema permite ejecutar experimentos de control con otros sistemas físicos, realizando para ello unos pequeños cambios en la configuración de RECOLAB. Esta es una de las ventajas de la arquitectura del sistema propuesto: su fácil ampliación y modificación.

De esta forma, el sistema físico localizado en el laboratorio (zona remota), está compuesto de los siguientes elementos:

- La unidad mecánica Feedback 33-002. Los componentes electromecánicos de esta unidad comprenden: un motor CC, un tacómetro analógico, potenciómetros analógico de señal de posición, encoders digitales incrementales absolutos y un freno magnético. En la figura 5 se muestra una imagen de esta maqueta junto con la cámara que permite enviar las imágenes adquiridas al servidor de vídeo.
- Tarjeta de adquisición de datos National Instruments 6024E. La tarjeta NI-6024E es una tarjeta de E/S analógica y digital de alta velocidad para computadores IBM PC compatibles. Algunas características interesantes de esta tarjetas son:
 - 16 canales (ocho diferenciales) de entrada analógica (12 bits de resolución)
 - 2 canales de salida analógica
 - 8 líneas de E/S digital
 - Frecuencia máxima de muestreo de 200 KS/s.

Esta tarjeta puede ser manejada directamente mediante Matlab /Simulink. El periodo de muestreo escogido por defecto es de $T=0.01$

segundos, pudiendo éste ser configurado. Dicho periodo es suficiente para realizar el control en tiempo real de los experimentos propuestos para el motor de CC.

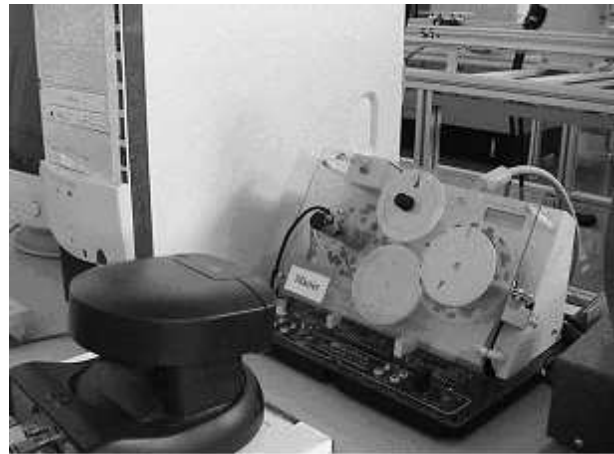


Figura 5. Servomotor Feedback 33-002

Por tanto una vez que el usuario ha experimentado con diferentes esquemas de control en simulación (acceso libre a todos los usuarios), puede realizar experimentos de control en la zona local con el sistema físico ubicado en la zona remota (para estos experimentos el usuario ha de estar previamente validado). Actualmente, los experimentos propuestos son los siguientes:

- Identificación del sistema.
- Diseño de un regulador PID en posición y velocidad.
- Diseño de reguladores de en el espacio de estados para control en posición y velocidad.

Estos experimentos permiten al usuario, ubicado en la zona local, utilizar los sistemas físicos del laboratorio sin requerir la presencia física en el mismo.

Cuando el usuario accede al sistema para realizar el control del sistema físico, el esquema Simulink que emplea varía ligeramente para acomodarse al sistema real (ver figura 6). De la misma forma que el sistema simulado, el usuario validado puede elegir el tipo de control a realizar, la selección del regulador y el valor de los parámetros de éste previamente a realizar el control del sistema real. Una vez enviados los parámetros seleccionados para el control del dispositivo físico, el usuario tiene la opción de visualizar el vídeo del sistema con la ejecución del sistema de control propuesto. Esta opción le permite al usuario acceder al servidor de vídeo instalado en el sistema. Finalizada la ejecución del experimento de control en tiempo real el usuario recibe el resultado de la misma tanto en formato gráfico como en fichero Matlab para su posterior análisis (ver figura 3).

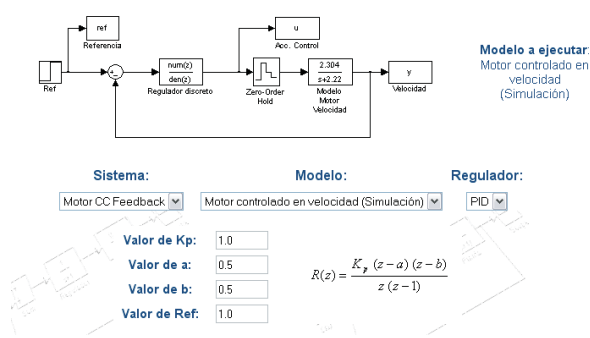


Figura 6. Esquema en Simulink para el control del sistema físico

4. USO DEL SISTEMA

El sistema RECOLAB se encuentra en funcionamiento desde inicio del curso académico 2003/04. La aceptación global del sistema por parte de los alumnos ha sido bastante aceptable. El número de accesos al sistema abierto a todos los usuarios ha sido muy elevado durante este primer curso académico (simulación de sistemas de control). Pero quizás más interés presenta el número y características de los accesos realizados a las partes del sistema que permiten un acceso al sistema de control en tiempo real.

De un total de 273 alumnos matriculados en las cuatro asignaturas de control en las que se imparten las prácticas en el laboratorio con este sistema físico, han accedido un 73 % de los mismos para realizar las prácticas a través de internet. Esto supone un elevado índice dado que durante el curso académico 2003/04 ha sido obligatorio la realización de las prácticas de forma presencial en el laboratorio ya que el sistema aún se encuentra en fase experimental. Estos datos permiten extraer como conclusión que la herramienta posibilita que los alumnos realicen de nuevo las prácticas que han desarrollado en el laboratorio con el sistema físico modificando los parámetros empleados así como los esquemas de regulación, o bien que completen las prácticas realizadas en el laboratorio y que por el limitado tiempo disponible no pueden ser terminadas en el horario asignado al efecto.

Otro dato claramente significativo del interés despertado en los alumnos con acceso restringido es que el promedio de acceso de cada uno de los usuarios al sistema RECOLAB ha sido de 5,3 ocasiones. Se observa pues que los usuarios prefieren acceder al sistema en diferentes sesiones aún cuando éstas no sean completas. Esta es otra de las ventajas del sistema para la realización de prácticas a través de internet, el usuario decide la duración de la sesión así como el número y la duración de las sesiones que desea emplear para experimentar con el sistema físico.

Los datos anteriormente expuestos permiten evidenciar la flexibilidad que puede suponer la herramienta para el estudiante proporcionándole una mayor disponibilidad temporal para realizar las tareas experimentales.

5. CONCLUSIONES

En este artículo se ha presentado una arquitectura capaz de realizar tareas de control en tiempo real a través de Internet. Asimismo, se permite un amplio intercambio de datos (gráficas, resultados, vídeo, etc.) entre el usuario y el laboratorio remoto. Para ello se ha empleado la plataforma Matlab/Simulink con los correspondientes toolboxes adicionales. El empleo de esta plataforma está justificada porque, por una parte, es una plataforma ampliamente utilizada en entorno de la ingeniería de control, y por otra, por su gran simplicidad y capacidad que presenta a la hora de desarrollar aplicaciones. En concreto, es preciso resaltar que tanto Matlab como Simulink son plataformas utilizadas en la mayor parte de las Escuelas de Ingeniería.

Aunque el ejemplo presentado en este artículo está basado en un motor CC, este mismo esquema de control en tiempo real a través de Internet, utilizando Matlab, puede ser aplicado a otros sistemas físicos disponibles en el laboratorio. El alumno puede configurar diversos parámetros del sistema de control entre diferentes valores predefinidos y precompilados (periodo de muestreo, regulador, etc.) de forma que se asegure la ejecución en tiempo real. El uso de Simulink permite añadir nuevos esquemas de control más complejos de forma off – line, exigiendo únicamente la compilación de los mismos mediante el Toolbox Real – Time Windows Target.

La principal ventaja del sistema propuesto es la facilidad que proporciona al alumno para realizar sus prácticas en teoría de control sobre laboratorios remotos. Asimismo, las ideas presentadas en este artículo proporcionan un método sencillo para validar nuevos esquemas de control, lo cual permite el estudio de diferentes algoritmos sobre diferentes sistemas físicos reales.

En todo caso, RECOLAB es un sistema en continuo desarrollo, habiéndose planteado mejoras en futuras versiones, que incluyen: la transmisión de datos en tiempo real para supervisión de variables mediante un nuevo cliente Java; la ampliación del número de prácticas y esquemas de control disponibles; la transmisión de video comprimido del experimento para poder trabajar con redes de ancho de banda limitado; así como la inclusión de tutoriales sobre teoría de control.

REFERENCIAS

- Apache (2004). The apache software foundation. <http://www.apache.org>.
- Ausserhofer, A. (1999). Web-based teaching and learning: A panacea?. *IEEE Communications Magazine* pp. 92–96.
- Bonivento, C., L.Gentili, L.Marconi and L.Rappini (2002). A web based laboratory for control engineering education. *Second International Workshop on Tele-Education in Engineering Using Virtual Laboratories*.
- Candelas, F.A., F. Torres, P. Gil, F. Ortiz, S. Puente and J. Pomares (2004). Laboratorio virtual remoto para robótica y evaluación de su impacto en la docencia. *Revista Iberoamericana de Automática e Informática Industrial*. pp. 49–57.
- Casini, M., D. Prattichizzo and A. Vicino (2003). The automatic control telelab: A user-friendly interface for distance learning. *IEEE Transactions on Education* **46**(3), 252–257.
- Díez, J.L., M. Vallés, A. Valera and J.L. Navarro (2001). Remote industrial process control with matlab web server. *IBCE 2001. IFAC Proceedings*.
- Dixon, W.E., D. Dawson, B. Costic and M. de Queiroz (2002). A matlab-based control systems laboratory experience for undergraduate students: Toward standardization and shared resources. *IEEE Transactions on Education* **45**(3), 218–226.
- Dixon, W.E., D.M. Dawson, B.T. Costic and M.S. de Queiroz (2001). Towards the standardization of a matlab-based control systems laboratory experience for undergraduate students. In: *Proceedings of the American Control Conference*. Arlington, VA. pp. 1161–1166.
- Exel, M., S. Gentil, F. Michau and D. Rey (2000). Simulation workshop and remote laboratory: two web-based training approaches for control. In: *Proc. American Control Conf.*. Chicago, IL, USA. pp. 3468–3472.
- Guimaraes, E., A. Maffei, R. Pinto, C. Miglinski, E. Cardozo, M. Bergerman and M. Bagalhaes (2003). Real- a virtual laboratory built form software components. In: *Proceedings of the IEEE*. Vol. 91.
- Hirukawa, H. and I. Hara (2000). Web-top robotics. *IEEE Robotics Automation Magazine* **7**, 40–45.
- Kleir, N.A., K. J. Astrom, D. Auslander, K.C. Cheok, G. F. Franklin, N. Masten and M. Rabins (1996). Control systems engineering education. *Automatica* **32**(2), 147–166.
- Mathworks (2004). Matlab r12, simulink, real – time workshop, real – time windows target and matlab web server. online manuals.. <http://www.mathworks.com>.
- Molly, H. Shor (2000). Remote – access engineering educational laboratoires: Who, what, when, why and how?. *Proceedings of the 2000 American Control Conference*. Chicago, IL.
- Molly, H. Shor and Robson Robby (2000). A student – centered feedback control model of the educational process. *30th ASEE/IEEE Frontiers in Education Conference*. Kansas City, MO.
- Overstreet, J. and A. Tzes (1999). An internet-based real-time control engineering laboratory. *IEEE Control Systems Magazine* **9**, 19–34.
- Pastor, R., Sánchez J. and Dormido S. (2001). Related: A framework for publish web laboratory control system. *IBCE 2001. IFAC Proceedings*. pp. 207–212.
- PHP (2004). Hypertext preprocessor. <http://www.php.net>.
- Puerto, R., L. Jiménez, O. Reinoso, C. Fernández and R. Neco (2004). Remote control laboratory using matlab and simulink: Application to a dc control model. *Ifac Workshop Internet Based Control Education*.
- Puerto, R., O. Reinoso, R.P. Neco, N.M. Garcia and L.M. Jimenez (2001). Remote lab for control applications using matlab. *Internet Based Control Education 2001. A proceedings volume from the IFAC Workshop Madrid* pp. 121–127.
- RECOLAB (2004). Portal web sistema recolab. <http://recolab.umh.es>.
- Schmid, C. (2001). Virtual control laboratories and remote experimentation in control engineering. *Proc. 11th EAAEIE Annual Conference on Innovations in Education for Electrical and Information Engineering, University of Ulm, Ulm* pp. 213–218.
- Sebastian, J.M., Garcia D. and Sanchez F.M. (2003). Remote-access education based on image acquisition and processing through the internet. *IEEE Transactions on Education* **46** (1).